Postprint version of paper published in Journal of Process Control 2010, vol. 20, pages 810-822.
The content is identical to the published paper, but without the final typesetting by the publisher.

Efficient multiple objective optimal control of dynamic systems with integer controls

F. Logist[a], S. Sager[b], C. Kirches[b], J.F. Van Impe[*,a]

[a] BioTeC & OPTEC, *Department of Chemical Engineering, Katholieke Universiteit Leuven, W. de Croylaan 46, B-3001 Leuven, Belgium*
[b] *Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, Im Neuenheimer Feld 368, 69120 Heidelberg, Germany*

Abstract

In practical optimal control problems both integer control variables and multiple objectives can be present. The current paper proposes a generic and efficient solution strategy for these multiple objective mixed-integer optimal control problems (MO-MIOCPs) based on deterministic approaches. Hereto, alternative scalar multiple objective optimisation techniques as normal boundary intersection and normalised normal constraint are used to convert the original problem into a series of parametric single objective optimisation problems. These single objective mixed-integer optimal control problems are then efficiently solved through direct multiple shooting techniques which exploit convex relaxations of the original problem. Moreover, these relaxations enable to quickly approximate the final solution to any desired accuracy (without the need of solving integer problems). Consequently, the set of Pareto optimal solutions of the MO-MIOCP can be accurately obtained in highly competitive computation times. The proposed method is illustrated on (*i*) a testdrive case study with a complex car model which includes different gears and conflicting minimum time - minimum fuel consumption objectives, and (*ii*) a jacketed tubular reactor case study with conflicting conversion, heat recovery and installation costs.

Key words: multiobjective optimization, mixed-integer optimization, optimal control, dynamic optimization, process control, automotive control

*Corresponding author: Tel.: +32-16-32.14.66 Fax: +32-16-32.29.91,
Email address: jan.vanimpe@cit.kuleuven.be (J.F. Van Impe)

Postprint of manuscript accepted for Journal of Process Control

1. Introduction

Many industrial processes can be accurately modelled by differential equations. Optimising their design and control, hence, gives rises to *dynamic optimisation* or *optimal control* problems, which have been studied extensively over the last 60 years (see, e.g., [1] for a historical review). However, much less results have been reported for specific subclasses, e.g., (*i*) *mixed-integer optimal control problems* (MIOCPs) with time-dependent control variables which can only take values from a finite set, or (*ii*) *multiple objective optimal control problems* (MOOCPs) with multiple and conflicting objective functions. This lack of generic results is not surprising since both classes are computationally challenging in nature. MIOCPs are closely related to the mixed-integer nonlinear programming problem (MINLP) class, which has been shown to be NP-hard [2]. MOOCPs typically give rise to a set of equally valid optimal solutions (i.e., the so-called *Pareto set*) instead of one single optimum [3]. Nevertheless, both classes are important for practical applications. MIOCPs are found, e.g., in car driving due to gear shifts [4, 5] or in process industry due to on-off valve switchings [6], while MOOCPs are encountered whenever trade-offs, e.g., between production and energy consumption, have to be accounted for (see, e.g., [7] for a review).

However, a lot of progress has been made over the last decade. For both problem classes generic and systematic solution approaches based on efficient deterministic procedures have been developed, resulting in a tremendous decrease in computation time. Driven by advances in scalar multiple objective optimisation (e.g., the development of generic deterministic approaches like *normal boundary intersection* (NBI) [8] and *normalised normal constraint* (NNC) [9]), procedures to quickly and efficiently generate the Pareto set for MOOCPs have lately been proposed by Logist et al. [10, 11]. For the mixed-integer optimal control cases, efficient convexification based direct multiple shooting approaches have recently been reported by Sager et al. [12]. An overview of (recent) advances in MIOCPs and references to the literature are provided in [13].

The aim of this paper is to design a generic, accurate and fast solution strategy for generating the Pareto set in MO-MIOCPs. The rationale behind the proposed strategy is a synergy between deterministic techniques from the fields of *multiple objective optimisation* (MOO) and *mixed-integer*

optimal control (MIOC). Here, the exploitation of deterministic convexification techniques for MIOCPs does not only allow to quickly approximate the exact MIOCP solution to any desired accuracy, but also enables a synergistic coupling with accurate deterministic reformulation approaches from continuous scalar MOO. In particular, scalarisation methods as NBI and NNC (which have been shown to mitigate the intrinsic drawbacks of the classic *weighted sum* (WS)) are used to convert the MO-MIOCP into a series of parametric single objective MIOCPs, while each of these MIOCPs is efficiently solved by a direct multiple shooting [14] approach, which exploits convex relaxations of the integer requirements. In summary, a synergistic effect is obtained between scalarisation methods which accurately yield the Pareto set for continuous scalar multiple objective optimisation problems (MOOPs), and convexification techniques which efficiently provide an accurate solution to the MIOCPs, resulting in highly competitive computation times. As a result, the Pareto optimal set can be computed up to any desired accuracy, without the need for solving integer problems. This is very important - while for stochastic approaches it is an advantage if the search space is limited to discrete values, for derivative-based deterministic approaches a discrete nature typically leads to an exponential increase in runtime. Hence, with the current approach the limitations of stochastic approaches can be overcome for an important problem class.

In Section 2 the mathematical formulation of a general MO-MIOCP is first introduced, then typical aspects and methods for MOO and MIOC are reviewed. Afterwards, the proposed approach for MO-MIOCPs is described. Section 3 introduces the case studies: (i) a testdrive case study which involves a detailed car model with gear shifts and exhibits conflicting minimum time - minimal fuel consumption objectives and (ii) a jacket tubular reactor for which only cooling fluid at certain temperatures is available in view of conflicting conversion, heat transfer and installation costs. The results are discussed in Section 4. Section 5 summarises the main conclusions.

2. Multiple-objective mixed-integer optimal control

In this section, first the general formulation of an MO-MIOCP is specified. Afterwards, specific concepts and existing approaches for scalar MOOPs and MIOCPs are reviewed. Finally, an alternative MO-MIOCP approach is presented. The rationale behind the proposed approach is that the convex

relaxation strategies exploited to quickly solve the MIOCPs at any desired accuracy (without the need for solving integer problems), enable also the coupling to accurate deterministic MOO procedures that generate the Pareto set in continuous scalar MOOPs.

2.1. General formulation

In the current study, an MO-MIOCP is defined as follows:

$$\min_{\mathbf{x}(\cdot),\mathbf{u}(\cdot),\mathbf{v}(\cdot),\mathbf{p},t_\mathrm{f}} \quad \{J_i(\mathbf{x}(\cdot),\mathbf{u}(\cdot),\mathbf{v}(\cdot),\mathbf{p},t_\mathrm{f})\}_{i=1\ldots m} \tag{1a}$$

$$\text{s.t.} \quad \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t),\mathbf{u}(t),\mathbf{v}(t)) \qquad \forall t \in \mathcal{T}, \tag{1b}$$

$$\mathbf{0} \leq \mathbf{c}_\mathrm{p}(\mathbf{x}(t),\mathbf{u}(t),\mathbf{v}(t),\mathbf{p}) \qquad \forall t \in \mathcal{T}, \tag{1c}$$

$$\mathbf{0} = \mathbf{r}_\mathrm{e}(\mathbf{x}(t_0),\mathbf{x}(t_\mathrm{f}),\mathbf{p},t_\mathrm{f}), \tag{1d}$$

$$\mathbf{0} \leq \mathbf{r}_\mathrm{i}(\mathbf{x}(t_0),\mathbf{x}(t_\mathrm{f}),\mathbf{p},t_\mathrm{f}), \tag{1e}$$

$$\mathbf{v}(t) \in \Omega \qquad \forall t \in \mathcal{T}. \tag{1f}$$

Let $t \in [t_0, t_\mathrm{f}] =: \mathcal{T} \subset \mathbb{R}$ be a fixed time horizon, and $\mathbf{x}(t) \in \mathbb{R}^{n_\mathrm{x}}$ describe the state vector of the dynamic process at any time $t \in \mathcal{T}$. Further, let $\mathbf{u}(t) \in \mathbb{R}^{n_\mathrm{u}}$ be the vector of continuous controls influencing the dynamic process, let $\mathbf{v}(t) \in \mathbb{R}^{n_v}$ be a vector of integer control functions, constrained to values from a discrete set $\Omega = \{v^1, v^2, \ldots, v^{n_w}\}$, and let $\mathbf{p} \in \mathbb{R}^{n_\mathrm{p}}$ be a vector with time-independent control functions. The vector $\mathbf{f}$ represents the dynamic system equations on the time interval $\mathcal{T}$. The vector $\mathbf{c}_\mathrm{p}$ contains path and control constraints on the time interval $\mathcal{T}$, including simple bounds. The vectors $\mathbf{r}_\mathrm{e}$ and $\mathbf{r}_\mathrm{i}$ indicate boundary equality and inequality constraints on the states. All functions are assumed to be sufficiently often differentiable. Note that this problem formulation can be generalised to include also algebraic variables, an explicit time-dependence, interior point constraints, multiple stages and so on. However, this would only complicate notation without additional insight, hence, the given specific formulation is concentrated on.

Each of the cost functions can be of the following type,

$$J_i(\mathbf{x}(\cdot),\mathbf{u}(\cdot),\mathbf{v}(\cdot),\mathbf{p},t_\mathrm{f}) = h_i(\mathbf{x}(t_\mathrm{f}),\mathbf{p},t_\mathrm{f}) + \int_{t_0}^{t_\mathrm{f}} g_i(\mathbf{x}(t),\mathbf{u}(t),\mathbf{v}(t),\mathbf{p})\mathrm{d}t, \tag{2}$$

consisting of a Mayer and a Lagrange term, for $1 \leq i \leq m$. The *feasible set* $\mathcal{U}$ is defined as the set of all admissible controls $(\mathbf{u}(\cdot),\mathbf{v}(\cdot),\mathbf{p},t_\mathrm{f})$, which induce

admissible state trajectories $\mathbf{x}(\cdot) \in \mathcal{X}$. Each element in $\mathcal{U}$ corresponds to a *cost vector* $\mathbf{J}(\mathbf{x}, \mathbf{u}, \mathbf{v}, \mathbf{p}, t_\mathrm{f}) = [J_1(\mathbf{x}, \mathbf{u}, \mathbf{v}, \mathbf{p}, t_\mathrm{f}), \ldots, J_m(\mathbf{x}, \mathbf{u}, \mathbf{v}, \mathbf{p}, t_\mathrm{f})]^\mathrm{T}$ and the set of all feasible cost vectors $\mathbf{J}$ yields the *feasible cost space* $\mathcal{J}_f$.

2.2. Multiple objective optimisation: concepts and methods

To illustrate several aspects of MOO, the MO-MIOCP (1) is simplified to a general scalar nonlinear MOOP:

$$\min_{\mathbf{y}} \{J_1(\mathbf{y}), \ldots, J_m(\mathbf{y})\} \quad \text{s.t.:} \quad \mathbf{y} \in \mathcal{S} \tag{3}$$

where the continuous decision variables $\mathbf{y}$ belong to the feasible set $\mathcal{S}$ and the vector of all individual cost functions is defined as $\mathbf{J}(\mathbf{y}) = [J_1(\mathbf{y}), \ldots, J_m(\mathbf{y})]^\mathrm{T}$.

Contrary to single objective optimisation (SOO), typically no single global solution exists in multiple objective optimisation. Therefore, it is necessary to determine a set of points that all fit a predetermined optimality definition, which is most often the concept of *Pareto optimality*.

Definition: *A point $\mathbf{y}^* \in \mathcal{S}$, is Pareto optimal iff there does not exist another point $\mathbf{y} \in \mathcal{S}$, such that $J_i(\mathbf{y}) \leq J_i(\mathbf{y}^*)$ for all i and $J_i(\mathbf{y}) < J_i(\mathbf{y}^*)$ for at least one objective function.*

In other words, a point is Pareto optimal if there exists no other feasible point that improves at least one objective function without worsening another.

Methods for generating the Pareto front are often classified into two classes: (*i*) *scalarisation methods* converting the MOO problem into a series of parametric single objective optimisation problems (SOOPs) (e.g., weighted sum, ...), and (*ii*) *vectorisation methods* tackling directly the MOO problem (e.g., stochastic evolutionary algorithms [15]). These vectorisation techniques are most often easily implemented, they can flexibly incorporate discrete variables, and they are generally regarded as global optimisation approaches. Unfortunately, these methods also exhibit certain restrictions. The repeated evaluation of the objectives is required (and, thus, also the repeated simulation of the underlying models), which can become time consuming for MO-MIOCPs. Constraints other than simple bounds on the decision variables are hard to cope with in an accurate way. And finally, due to the

stochastic nature of the search procedures, the dimension of the search space has to be kept rather low (resulting in very crude control discretisations for MO-MIOCPs). On the other hand, scalarisation techniques in general do not suffer from these drawbacks since efficient deterministic optimisation approaches can be exploited to find a (local) optimum for the different SOOPs (which can be large-scale). However, the efficient solution of these SOOPs is crucial, because their number increases exponentially with the number of the objectives. Nevertheless, in the current study, only deterministic scalarisation MOO techniques will be exploited, since the aim is to integrate them with efficient deterministic approaches for solving MIOCPs.

2.2.1. Weighted Sum (WS)

The most often employed deterministic reformulation technique in practice is combining the different objectives into a convex weighted sum, resulting in the following parametric SOO problem:

$$\min_{\mathbf{y}\in\mathcal{S}} J_{\text{ws}} = \sum_{i=1}^{m} w_i J_i(\mathbf{y}) \text{ with } w_i \geq 0 \text{ and } \sum_{i=1}^{m} w_i = 1. \tag{4}$$

By consistently varying the weight vector $\mathbf{w} = [w_1, w_2, \ldots, w_i, \ldots, w_m]^{\text{T}}$ an approximation of the Pareto set is obtained. However, despite its simplicity, the weighted sum approach has several intrinsic drawbacks [16]. A uniform distribution of the weight vector does not necessarily result in an even spread on the Pareto front and points in non-convex parts of the Pareto set cannot be obtained.

2.2.2. Normal Boundary Intersection (NBI)

This method has been proposed by Das and Dennis [8] to mitigate the above mentioned drawbacks of the WS. NBI tackles the MOO problem from a geometrically intuitive viewpoint. It first builds a plane in the cost space $\mathcal{J}_f$ which contains all convex combinations of the individual minima, i.e., the *convex hull of individual minima* (CHIM), and then constructs *(quasi-)normal* lines to this plane. The rationale behind the method is that the intersection between the (quasi-)normal from any point $\mathbf{J}_p$ on the CHIM, and the boundary of the feasible cost space closest to the origin is expected to be Pareto optimal. Hereto, the MOO problem is reformulated as to maximise the distance λ from a point $\mathbf{J}_p$ on the CHIM along the quasi-normal through this point, without violating the original constraints. Technically,

Postprint version of paper published in Journal of Process Control 2010, vol. 20, pages 810-822.
The content is identical to the published paper, but without the final typesetting by the publisher.

this requirement of lying on the quasi-normal introduces additional equality constraints, resulting in the following formulation:

$$\max_{\mathbf{y}\in\mathcal{S},\lambda} \lambda \quad \text{s.t.:} \quad \mathbf{\Phi w} - \lambda \mathbf{\Phi e} = \mathbf{J}(\mathbf{y}) - \mathbf{J}^* \tag{5}$$

where $\mathbf{\Phi}$ is the $m \times m$ pay-off matrix in which the i-th column is $\mathbf{J}(\mathbf{y}_i^*) - \mathbf{J}^*$. $\mathbf{y}_i^*$ is the minimiser of the i-th objective J_i and $\mathbf{J}^*$ is the *utopia point*, which contains the minima of the individual objectives $J_i(\mathbf{y}_i^*)$. $\mathbf{w}$ is again a vector of 'weights' $\mathbf{w} = [w_1, w_2, \ldots, w_i, \ldots, w_m]^{\mathrm{T}}$ such that $\sum_{i=1}^{m} w_i = 1$ with $w_i \geq 0$, and $\mathbf{e}$ is a vector containing all ones. Now, $\mathbf{\Phi w}$ describes a point in the CHIM and $-\mathbf{\Phi e}$ defines the (quasi-)normal to the CHIM pointing towards the origin. When the points on the CHIM are selected with an equal spread (via a uniform distribution of the 'weight' vectors $\mathbf{w}$), also an equal spread on the Pareto frontier in the cost space is obtained.

2.2.3. Normalised Normal Constraint (NNC)

NNC, as introduced by Messac et al. [9], employs similar ideas as NBI, but combines them with features of the ε-constraint method [17]. This ε-constraint method minimises the single most important objective function J_k, while the $m-1$ other objective functions are added as inequality constraints $J_i \leq \varepsilon_i$. These inequalities can be interpreted as hyperplanes reducing the feasible cost space. After normalisation of the objectives, NNC also first constructs a plane through all individual minima (called here, the *utopia hyperplane*). Then NNC minimises a selected (normalised) objective $\overline{J}_k$, given the original constraints, and while additionally reducing the feasible space by adding $m-1$ hyperplanes through a selected point $\overline{\mathbf{J}}_p$ in the utopia plane. These hyperplanes are chosen perpendicular to each of the $m-1$ *utopia plane vectors*, which join the individual minimum $\overline{\mathbf{J}}(\mathbf{y}_k^*)$ corresponding to the selected objective $\overline{J}_k$, with all other individual minima $\overline{\mathbf{J}}(\mathbf{y}_i^*)$. Hence, this approach leads to an additional set of inequality constraints:

$$\min_{\mathbf{y}\in\mathcal{S}} \overline{J}_k \quad \text{s.t.:} \quad (\overline{\mathbf{J}}(\mathbf{y}_k^*) - \overline{\mathbf{J}}(\mathbf{y}_i^*))^{\mathrm{T}} (\overline{\mathbf{J}}(\mathbf{y}) - \overline{\mathbf{J}}_{\mathrm{p}}) \leq \mathbf{0}, \quad i = 1 \ldots m, i \neq k. \tag{6}$$

As in NBI, evenly distributed points on the utopia plane $\overline{\mathbf{J}}_p$ can be selected by a uniform variation of a 'weight' vector $\mathbf{w} = [w_1, w_2, \ldots, w_i, \ldots, w_m]^{\mathrm{T}}$ (with $\sum_{i=1}^{m} w_i = 1$ and $w_i \geq 0$), which also ensures an even spread on the Pareto set. To guarantee a scale independent solution for all number of objectives,

Journal homepage: http://www.sciencedirect.com/science/journal/09591524
Original file available at: http://dx.doi.org/10.1016/j.jprocont.2010.04.009

the enhancement reported in [18] has been exploited. A geometric interpretation of NBI and NNC for a bi-objective case is presented in Figure 1.

Remarks:

- Since NBI and NNC may return non-Pareto optimal points in a limited number of situations, a Pareto filter algorithm has been applied afterwards. The implemented Pareto filter (adopted from [9]) is based on a pairwise comparison of the candidate Pareto optimal solutions and the removal of dominated (non-Pareto optimal) ones.

- When more than two objectives are present, some Pareto optimal solutions (especially near the boundaries) may be overlooked by NBI and NNC. This phenomenon is related to the fact that only convex weights ($w_i \geq 0$) are taken. Hence, this inability has been countered by removing the positivity constraint on the weights. However, to avoid the solution of unnecessary SOOPs, weight generating procedures similar to the one mentioned in [19] have been exploited. Although the procedure removes the positivity requirement for w_i, the set of possible weight vectors is limited based on geometric grounds, i.e., no weight vector should be tried that can only lead to infeasible or dominated solutions.

2.2.4. Algorithm

Hence, the above mentioned reformulation/scalarisation strategies for MOOPs can be schematically represented by the following algorithm.

Algorithm 2.1. (MO Scalarisation)

1. *If necessary, compute the individual minima* $J_i(\mathbf{y}_i^*)$ *of all objectives given the original constraints* $\mathbf{y} \in \mathcal{S}$*, and normalise all objectives to* $\overline{J}_i(\mathbf{y})$.
2. *Input: a finite set of 'weight' vectors* $\mathbf{w}^k$*,* $k = 0, \ldots, n_{w_k}$*, containing the 'weights' for the* n_{w_k} *different Pareto optimal points to be generated. Since no prior knowledge is available, most often a uniform distribution of the 'weight' vectors is adopted. (For cases with more than two objectives and as reformulation method NBI or NNC, the adapted weight generation procedure [19], which leaves out the positivity requirement* $w_i \geq 0$ *may have to be employed in order to avoid the overlooking of Pareto points near the boundaries.)*

Postprint version of paper published in Journal of Process Control 2010, vol. 20, pages 810-822.
The content is identical to the published paper, but without the final typesetting by the publisher.

3. *Reformulate the original MOO (Equation (3)) to a parametric SOO (with the weights as parameters to be varied) via WS (Equation (4)), NBI (Equation (5)), or NNC (Equation (6)).*
4. *FOR $k = 0$ to n_{w_k}:*
   (a) *Solve the parametric SOO problem for the 'weights' vector $\mathbf{w}^k$ using an appropriate NLP optimisation routine.*
   (b) *Compute the cost vector $\mathbf{J}(\mathbf{y}^k)$ for the optimised decision variables $\mathbf{y}^k$. (If $\mathbf{J}(\mathbf{y}^k)$ happens to dominate one of the individual minima (e.g., due to the existence of local minima), restart and employ $\mathbf{J}(\mathbf{y}^k)$ as one of the individual minima instead of the dominated one.)*
5. *END*
6. *If necessary, apply a Pareto filter algorithm (see, e.g., [9]) to filter out non-Pareto optimal points from the obtained set of points in the cost space $\mathbf{J}(\mathbf{y}^k)$.*

2.3. Mixed-integer optimal control: concepts and methods

It is imperative for the successful application of the MOO scalarisation approaches presented in the previous section that the parametric SOOPs be solved efficiently. In this section, a MIOCP of the form described in Equation (1) is considered, however with a single objective $J(\cdot)$ instead of the set given by Equation (1a) (i.e., $m = 1$). Adequate problem formulations and resulting theoretical insights are discussed, as well as methods to calculate optimal integer controls.

2.3.1. Outer convexification

In the remainder, the term *integer control* will be used for Equation (1f), while *binary control* refers to the special case $\omega(t) \in \{0, 1\}^{n_\omega}$. The expression *relaxed* will be employed, whenever a restriction $\mathbf{v}(\cdot) \in \Omega$ is relaxed to a convex control set with a recently proposed outer convex relaxation [20] that is defined as follows. For every element $\mathbf{v^i}$ of Ω a binary control function $\omega_i(\cdot)$ is introduced. The ODE system described by Equation (1b) can then be written as Equation (7b). If an additional special ordered set type one

Journal homepage: http://www.sciencedirect.com/science/journal/09591524
Original file available at: http://dx.doi.org/10.1016/j.jprocont.2010.04.009

condition (Equation (7f)) is imposed, the MIOCP is obtained:

$$\min_{\mathbf{x}(\cdot),\mathbf{u}(\cdot),\mathbf{v}(\cdot),\mathbf{p},t_\mathrm{f}} \quad J(\mathbf{x}(\cdot),\mathbf{u}(\cdot),\mathbf{v}(\cdot),\mathbf{p},t_\mathrm{f}) \tag{7a}$$

$$\text{s.t.} \quad \dot{\mathbf{x}}(t) = \sum_{i=1}^{n_\omega} \mathbf{f}(\mathbf{x}(t),\mathbf{u}(t),\mathbf{v}^\mathbf{i},\mathbf{p})\,\omega_i(t) \qquad \forall t \in \mathcal{T}, \tag{7b}$$

$$\mathbf{0} \leq \mathbf{c}_\mathrm{p}(\mathbf{x}(t),\mathbf{u}(t),\mathbf{v}(t),\mathbf{p}) \qquad \forall t \in \mathcal{T}, \tag{7c}$$

$$\mathbf{0} = \mathbf{r}_\mathrm{e}(\mathbf{x}(t_0),\mathbf{x}(t_\mathrm{f}),\mathbf{p},t_\mathrm{f}), \tag{7d}$$

$$\mathbf{0} \leq \mathbf{r}_\mathrm{i}(\mathbf{x}(t_0),\mathbf{x}(t_\mathrm{f}),\mathbf{p},t_\mathrm{f}), \tag{7e}$$

$$1 = \sum_{i=1}^{n_\omega} \omega_i(t) \qquad \forall t \in \mathcal{T}, \tag{7f}$$

$$\omega(t) \in \{0,1\}^{n_\omega} \qquad \forall t \in \mathcal{T}. \tag{7g}$$

There is a bijection between every feasible integer function $\mathbf{v}(\cdot) \in \Omega$ of the MIOCP described by Equation (1) (with m=1), and an appropriately chosen binary function $\omega(\cdot) \in \{0,1\}^{n_\omega}$ for Equation (7), see [20]. The relaxation of $\omega(t) \in \{0,1\}^{n_\omega}$ is given by $\alpha(t) \in [0,1]^{n_\omega}$.

This formulation has two main advantages. First, for many OCPs the optimal solution will have a *bang–bang* character, therefore the solution of the relaxed problem will yield the optimal integer solution. Second, for problems that fit into the class a theory has been developed that allows to deduce information on the optimal integer solution from the optimal value of the relaxed problem, even if this solution is not *bang–bang*, but *path-constrained* or *sensitivity-seeking*. A constructive way to obtain such a solution in polynomial time has been found, i.e., the *Sum Up Rounding* strategy, described in [20]. This theory is the driving force behind the current method to calculate integer solutions in an error-controlled way.

2.3.2. Calculation of integer solutions

As stated before, a direct, multiple shooting based approach will be used to solve continuous optimal control problems. Different methods for the calculation of integer solutions for MIOCPs, based on a direct approach, have been described and compared in [20], e.g., Branch&Bound, Outer Approximation, Penalisation Heuristics and Rounding Strategies. All methods that suffer from a combinatorial explosion when the number of discretised binary

control variables increases have a very limited applicability, though.

It can often be observed that the solution of the relaxed, purely continuous problem already yields an integer solution for almost all control discretisations. In addition, simple rounding strategies, taking the special ordered set constraint (Equation (7f)) into account, often result in integer solutions without affecting the objective function value (and the underlying solution).

For cases in which path constraints play a role or a different objective function leads to sensitivity-seeking arcs, it is recommended to use a *Sum Up Rounding* strategy, possibly in combination with a *Switching Time Optimisation* approach, as developed in [20, 12]. Sum Up Rounding yields integer solutions arbitrarily close to the optimal integer solution, if a *sufficiently fine* time discretisation is used. If guaranteed global solutions are an issue, this approach can be readily combined with methods in global optimisation, of course.

2.3.3. Algorithm

Hence, the following algorithm is proposed for the solution of single objective MIOCPs (i.e., Equation (1) with $m = 1$). Here, the control discretisation grid in iteration k is denoted with $\mathcal{G}^k$, and the optimal trajectory of this single objective MIOCP with $\mathcal{T}^k = (x^k(\cdot), u^k(\cdot), \alpha^k(\cdot))$.

Algorithm 2.2. (MS MINTOC)

1. $k = 0$. *Input: control discretisation grid* $\mathcal{G}^0$, *tolerance* $TOL \in \mathbb{R}^+$.
2. *If necessary, reformulate and convexify (Section 2.3.1) the single objective MIOCP (i.e., Equation* (1) *with m=1).*
   Obtain problem of type described by Equation (7). *Relax the control* $\omega(\cdot)$ *to* $\alpha(\cdot) \in [0,1]^{n_\omega}$.
3. *REPEAT*
   (a) *Solve relaxed problem on* $\mathcal{G}^k$. *Obtain* $\mathcal{T}^k = (x^k(\cdot), u^k(\cdot), \alpha^k(\cdot))$ *and the grid-dependent optimal value* $\Phi^{REL}_{\mathcal{G}^k}$.
   (b) *If* $\mathcal{T}^k$ *on* $\mathcal{G}^k$ *fulfils* $\omega^k(\cdot) := \alpha^k(\cdot) \in \{0,1\}^{n_\omega}$ *then STOP.*
   (c) *Apply Sum Up Rounding [20] to* $\alpha^k(\cdot)$. *Fix* $u^k(\cdot)$.
   Obtain $y^k(\cdot)$ *and upper bound* $\Phi^{BIN}_{\mathcal{G}^k}$ *by simulation.*
   (d) *If* $\Phi^{BIN}_{\mathcal{G}^k} < \Phi^{REL}_{\mathcal{G}^k} + TOL$ *then STOP.*
   (e) *Refine the control grid* $\mathcal{G}^k$.

(f) $k = k + 1$.

4. *Bijection to obtain solution for the single objective MIOCP (i.e., Equation* (1) *with m=1) with objective* $\Phi^* = \Phi^{BIN}_{\mathcal{G}^k}$.

As for all algorithms the question has to posed whether it is well-posed and will terminate in a finite number of steps. The answer is given by the following theorem.

Theorem 2.3. (Well-posedness of MS MINTOC)
If the assumptions

1. *On all grids* $\mathcal{G}^k$ *an optimal solution to the relaxed problem of the one described by Equation* (1) *(with* $m = 1$*) is found in a finite number of operations.*
2. *Bisection is used for the refinement of* $\mathcal{G}^k$.
3. *After a finite number* k^{max} *of refinements the optimal relaxed solution is frozen,* $\mathcal{T}^k = \mathcal{T}^{k^{max}}$ *and* $\Phi^{REL}_{\mathcal{G}^k} = \Phi^{REL}_{\mathcal{G}^{k^{max}}} \; \forall \; k > k^{max}$.

hold, then Algorithm 2.2 will terminate in a finite number of steps with a feasible binary solution, for which $\Phi^* < \Phi^{REL}_{\mathcal{G}^k} + TOL$ *holds.*

The proof is given in [13]. The main advantage of Algorithm 2.2 is obviously that only continuous control problems without the need for a combinatorial search need to be solved, plus application of the Sum Up Rounding strategy for which the effort is negligible.

2.4. An efficient and generic approach for MO-MIOCPs

The proposed approach for MO-MIOCPs integrates (*i*) the *accurate MOO reformulation strategies NBI and NNC* with (*ii*) the *fast direct multiple shooting methods* which exploit convex relaxations for MIOCPs. The following algorithm schematically describes the combination of the MOO and MIOCP approaches presented so far. The rationale behind the approach is that, due to synergistic effects, the Pareto optimal set can be approximated to any desired accuracy without the need for solving integer problems.

Algorithm 2.4. (MO-MINTOC)

1. *Outer loop: apply WS, NBI or NNC to convert the MO-MIOCP to a series of parametric single objective MIOCPs (Algorithm 2.1).*
2. *Determine control discretisation grid* $\mathcal{G}^k$

3. *Inner loop:*
   (a) *If necessary, reformulate and convexify (see Section 2.3.1) the single objective MIOCP* (1)*. Obtain problem of type* (7)*. Relax the control* $\omega(\cdot)$ *to* $\alpha(\cdot) \in [0,1]^{n_\omega}$.
   (b) *Initialise variables, preferably by using hot-starts.*
   (c) *Solve relaxed problem on* $\mathcal{G}^k$ *and obtain the grid- and weight-dependent optimal value* $\Phi^{REL}_{\mathcal{G}^k}$.
4. *Analyse Pareto front, choose prefered setpoint*
5. *Calculate integer solution from relaxed solution using Algorithm 2.2*

The proposed approach has been implemented based on the software packages `MS-MINTOC` [20] and `MUSCOD-II` [21, 22]. Due to the deterministic nature and the convex relaxations, highly accurate results (e.g., fine control parameterisations, incorporation of path and terminal constraints, ...) can be obtained in very competitive computation times. It should be emphasised that the integer controls are calculated only a posteriori, typically in negligible runtime compared to the overall runtime. The resulting integer controls and the corresponding objective function vector depend on the requested tolerance TOL in Algorithm 2.2 and 2.4. However, as TOL is decreased, the difference between the objective function vector corresponding to the integer feasible controls and the relaxed Pareto point also decreases, and can -in principle- be reduced to zero. Hence, the relaxed Pareto set can be approximated by integer feasible solutions to any desired accuracy. However, the price to be paid may be the number of (*i*) switchings between different integer values and/or (*ii*) Sum Up Rounding steps.

Algorithm 2.4 is generic and can easily be applied to any optimisation problem of type (1). As there are no further assumptions on functions, the optimisation problem may be non-convex. If local optimisation methods are used to solve the relaxed control problems, only local (Pareto) optimality can be ensured, although simultaneous optimal control approaches as collocation or multiple shooting may help to avoid certain bad local minima.

The determination of the control discretisation grid $\mathcal{G}^k$ can be automated, by checking improvements in the objective compared to an extrapolation. However, special care needs to be taken as this grid will be applied to all of the single objective optimisation problems.

Although not mentioned explicitly, the current approach is also applicable to cases with time-invariant discrete controls. However, in this case, an additional loop (with, e.g., a Branch&Bound approach) will be required for each of the parametric MIOCPs, resulting in an increase in computational burden.

3. Case studies

To test the proposed MO-MIOCP approach, two cases are studied: (*i*) a testdrive of car and (*ii*) the design of a jacketed tubular reactor, which exhibit two and three conflicting objectives, respectively.

3.1. Car testdrive

A time-optimal car driving manoeuvre to avoid an obstacle with small steering effort is considered. At any time, the car must be positioned on a prescribed track. This control problem was first formulated in [4] and used for subsequent studies [5, 23]. Here, two conflicting objectives are adopted: minimising time versus minimising consumed fuel.

3.1.1. Car model and constraints

The considered car model is derived under the simplifying assumption that rolling and pitching of the car body can be neglected. Only a single front and rear wheel is modelled, located in the virtual center of the original two wheels. Motion of the car body is considered on the horizontal plane

only. This results in the following ODE system for $t \in [t_0, t_f]$:

$$\begin{aligned}
\dot{c}_x &= v\cos(\psi-\beta) && (8)\\
\dot{c}_y &= v\sin(\psi-\beta) && (9)\\
\dot{v} &= \frac{1}{m}\Big((F_{lr}^{\mu}-F_{Ax})\cos\beta + F_{lf}\cos(\delta+\beta) \\
&\quad -(F_{sr}-F_{Ay})\sin\beta - F_{sf}\sin(\delta+\beta)\Big) && (10)\\
\dot{\delta} &= w_\delta && (11)\\
\dot{\beta} &= w_z - \frac{1}{mv}\Big((F_{lr}-F_{Ax})\ \sin\beta + F_{lf}\ \sin(\delta+\beta) \\
&\quad +(F_{sr}-F_{Ay})\ \cos\beta + F_{sf}\ \cos(\delta+\beta)\Big) && (12)\\
\dot{\psi} &= w_z && (13)\\
\dot{w}_z &= \frac{1}{I_{zz}}\big(F_{sf}\,l_f\ \cos\delta - F_{sr}\,l_r - F_{Ay}\,e_{SP} + F_{lf}\,l_f\ \sin\delta\big) && (14)
\end{aligned}$$

with as initial and terminal conditions:

$$\begin{aligned}
\mathbf{x}(t_0) &= [-30, \text{free}, 10, 0, 0, 0, 0]^T && (15)\\
\mathbf{x}(t_f) &= [140, \text{free}, \text{free}, \text{free}, \text{free}, 0, \text{free}]^T && (16)
\end{aligned}$$

and constraints:

$$\begin{aligned}
w_\delta(t) &\in [-0.5, 0.5], \quad F_B(t) \in [0, 1.5\cdot 10^4], \quad \phi(t) \in [0,1] && (17)\\
\mu(t) &\in \{1,\ldots,5\} && (18)\\
c_y(t) &\in \left[P_l(c_x(t)) + \tfrac{B}{2}, P_u(c_x(t)) - \tfrac{B}{2}\right] && (19)\\
v(t) &\in \left[\frac{800\pi R}{30 i_i i_g^{\mu}}, \frac{8000\pi R}{30 i_i i_g^{\mu}}\right] && (20)\\
t_f &\le 12.0 && (21)
\end{aligned}$$

The differential states contained in $\mathbf{x}(\cdot)$ are the horizontal position of the car c_x, the vertical position of the car c_y, the magnitude of directional velocity of the car v, the steering wheel angle δ, the side slip angle β, the yaw angle ψ, and the yaw angle velocity w_z. The three continuous control functions contained in $\mathbf{u}(\cdot)$ are the steering wheel angular velocity w_δ, the total braking force F_B and the accelerator pedal position ϕ, while the discrete

control vector $\mathbf{v}(\cdot)$ consists of the gear choice μ.

Bounds are imposed on the continuous and discrete control variables (Equations (17) and (18)), while highly nonlinear state constraints originate due to the track and bounds on the engine velocity (Equations (19) and (20)). The model parameters, the forces and expressions in Equations (8) - (21), and the parameterisation of the track depicted in Figure 9 can be found in either one of [4, 5, 23, 24].

3.1.2. Objectives

Two objectives are considered: minimising the *fuel consumption*, which is assumed to be measurable via the total number of engine rotations [100 engine rotations]:

$$J_1 = \frac{1}{100} \int_0^{t_\mathrm{f}} \frac{30 i_\mathrm{t} i_\mathrm{g}^\mu}{\pi R} v(t) \mathrm{d}t, \tag{22}$$

and minimising the *travelling time* [s]:

$$J_2 = t_\mathrm{f}. \tag{23}$$

In J_1 a scaling factor of 100 has been added in order to bring both costs to the same order of magnitude. As can easily be seen, these objectives are incommensurable and even conflicting.

3.2. Jacketed tubular reactor design

The studied reactor involves a tubular chemical reactor operating under steady-state conditions. Inside the reactor an exothermic reaction takes place, while a surrounding jacket enables the heat removal. Three conflicting objectives are selected: maximising conversion, maximising heat recovery and minimising the installation cost.

3.2.1. Reactor model and constraints

The reactor model adopted is based on the 1D plug flow model from [25]. It is based on the following assumptions: (*i*) no axial dispersion, (*ii*) steady-state conditions, (*iii*) perfect radial mixing, (*iv*) a constant density and heat capacity of the fluid, (*v*) a negligible heat resistance between the reactor and its jacket, and (*vi*) an Arrhenius law dependence of the reaction

rate on the temperature. Using the spatial coordinate along the reactor z as the independent variable yields an ODE system for $z \in [0, L]$:

$$\frac{dx_1}{dz}(z) = \frac{\alpha}{v_\mathrm{f}}(1 - x_1)e^{\frac{\gamma x_2}{1+x_2}} \tag{24}$$

$$\frac{dx_2}{dz}(z) = \frac{\alpha\delta}{v_\mathrm{f}}(1 - x_1)e^{\frac{\gamma x_2}{1+x_2}} + \frac{\beta}{v_\mathrm{f}}(v - x_2) \tag{25}$$

with initial conditions:

$$\mathbf{x}(0) = [0, 0]^\mathrm{T} \tag{26}$$

and constraints:

$$x_2(z) \in \left[\frac{T_\mathrm{min} - T_F}{T_F}, \frac{T_\mathrm{max} - T_F}{T_F}\right] \tag{27}$$

$$v(z) \in \left\{\frac{T_{w,\mathrm{min}} + i\Delta T_w - T_F}{T_F}\right\}_{i=0,1,\ldots,4} \text{with } \Delta T_w = \frac{T_{w,\mathrm{max}} - T_{w,\mathrm{min}}}{4} \tag{28}$$

$$L \in [0.4, 1] \tag{29}$$

$$x_1(L) \geq 0.85 \tag{30}$$

Here, the state vector $\mathbf{x}(\cdot)$ contains the dimensionless versions of the concentration $x_1 = \frac{C_\mathrm{F} - C}{C_\mathrm{F}}$ and the reactor temperature $x_2 = \frac{T - T_\mathrm{F}}{T_\mathrm{F}}$, with T_F and C_F the temperature and reactant concentration of the feed stream, respectively. The discrete control vector $\mathbf{v}(\cdot)$ contains $v = \frac{T_w - T_\mathrm{F}}{T_\mathrm{F}}$ the dimensionless version of the jacket temperature T_w, which is only available at five distinct temperature levels. This integer restriction has to be seen in a plant-wide context, where cooling fluid may only be available at certain temperature levels (e.g., from other production units or from general utility units) and where it is not desirable to spend additional effort on preliminary heat exchangers to get temperatures in between.

Bounds are imposed on the reactor and jacket temperatures, as well as on the reactor length (Equations (27) - (29)) for constructive reasons, while Equation (30) ensures that a minimum conversion is achieved. Expressions and parameter values can be found in [25].

416 *3.2.2. Objectives*

Three objectives are selected: maximising the *conversion*, which is related to minimising the reactant concentration at the outlet [mole/L]:

$$J_1 = C_F(1 - x_1(L)), \tag{31}$$

maximising the net *heat transfer* between the reactor and its jacket, where heat transfered from the reactor to the jacket is assumed to a profit [1/s]:

$$J_2 = \int_0^L \frac{\beta}{L}(v - x_2(z))dz, \tag{32}$$

and minimising the *installation cost*, which is linked to the reactor length [m]:

$$J_3 = L. \tag{33}$$

417 The first two objectives are similar to the conflicting ones treated in [26].
418 However, the third one, that is added in the current study, clearly counter-
419 acts the previous two, since shorter reactors give rise to lower conversions
420 and net heat transfers.
421

422 4. Results and discussion

423 In this part the results obtained with the scalarisation based approaches
424 for the double-lane change manoeuvre and the jacketed tubular reactor are
425 presented in Sections 4.1 and 4.2, respectively. Here, it will be shown that
426 the Algorithm 2.4 allows to compute the Pareto set to any desired accuracy
427 without the need of actually solving integer problems. In Section 4.3 - when-
428 ever possible - a comparison is made with a classic stochastic evolutionary
429 MOO approach, i.e., the `NSGA-II` algorithm [27]. Afterwards in Section 4.4,
430 an indication of the computational expense is given. Finally, Section 4.5
431 comments on the satisfaction of state constraints in direct optimal control
432 approaches.
433

434 Concerning the reconstruction of integer solutions from the relaxed ones,
435 (i) structure-exploiting rounding strategies (taking the special ordered set
436 constraint (1f) into account) and (ii) switching time optimisation can be
437 applied. However, in the context of the current multi-objective optimisation

study, the primal interest is in obtaining the Pareto front. Only when an adequate setpoint is identified (by the decision maker), the issue of how to obtain an integer solution arises. Nevertheless, such solutions are shown for illustration in Figures 4 and 7.

4.1. Testdrive: Pareto set and optimal trajectories

The Pareto front is computed for a uniform grid with 11 points. Hence, the 'weight' vector is each time constructed as $\mathbf{w} = [1-w, w]^{\mathrm{T}}$ for w going from 0 to 1 in steps of 0.1. To obtain the objective function value of the resulting MIOCPs, the outer convexification formulation is exploited. All control functions are discretised by piecewise constant functions on 80 equidistant intervals. The integration and optimality tolerances employed are 10^{-8} and 10^{-10}, respectively. For comparison reasons, the same initial guess has been used for all parametric SOOPs. In practice however, important computational gains can be achieved with hot-starts.

The Pareto sets obtained with the different scalarisation methods based on the convex relaxation are depicted in Figure 2. (Since NBI and NNC yield exactly the same results, only the NBI results are displayed.) The time and fuel consumption costs are clearly conflicting as a decrease in travelling time has to be compensated by an increase in fuel consumption. Nevertheless, these inherent trade-offs occur in a *natural* way as the Pareto frontier is convex (if no active set changes occur, compare Section 4.5). Clearly the NBI and NNC results exhibit a more or less uniform spread along the Pareto set, whereas the results obtained with the WS are unevenly spread (despite the scaling in the first objective). When looking at the extreme points, it should be noted that for the pure minimum time optimal control problem (i.e., $\mathbf{w} = [0, 1]^{\mathrm{T}}$), the same results as reported by Gerdts [4] are obtained ($J_2 = 6.78628$ s for a control discretisation with 40 intervals), while for the pure minimum fuel case (i.e., $\mathbf{w} = [1, 0]^{\mathrm{T}}$), the upper time limit is reached ($J_2 = 12.0000$ s).

A selection of the corresponding state and control trajectories based on NBI are displayed in Figure 3 for different points along the Pareto set. The top left plot in Figure 3 illustrates the trajectory driven by the car. Apart for the pure minimum fuel case (i.e., $\mathbf{w} = [1, 0]^{\mathrm{T}}$), hardly any differences are visible. The different driving trajectory for $\mathbf{w} = [1, 0]^{\mathrm{T}}$ is induced by the lower velocity, allowing to take sharper turns and, hence, minimising total distance

to be covered. The other plots in Figure 3 depict the velocity v, the gear choice μ and the engine speed n_{eng} as a function of the horizontal position. In all cases there is an increase in velocity as the car moves to the right. This observation is also due to the fact that (i) the accelerator is in all cases fully pushed down during the entire interval and (ii) the brake is (almost) never hit. However, as could be expected, the increase in velocity is higher when more emphasis is put on the time cost (i.e., when w increases). In these last cases also low gears are preferred. As can be seen, the gears are gradually changed up. For instance, for the pure time optimal case (i.e., $\mathbf{w} = [0, 1]^{\mathrm{T}}$), the car starts in first gear, shifts to second after $\pm$ 5 m, performs the two lane changes in third gear, and finally, changes up to fourth after arriving back onto the starting line in a straight position. However, when fuel consumption is stressed, this gear shifting occurs earlier. In the extreme case of minimum fuel consumption, the car immediately changes to fifth gear. This behaviour is easily explained based on the engine speed trajectories. Since higher gears induce lower engine speeds at the same velocity, higher gears are preferred when the minimisation of the fuel consumption (measured by the total number of engine rotations) is focused on. As expected, a gradual increase in the engine speed trajectories is observed, when w is increased. Clearly, when integer integer feasible solutions are computed afterwards (Figure 4) hardly differences are visible.

4.2. Tubular reactor: Pareto set and optimal trajectories

As in the testdrive case, the step in the 'weights' is taken equal to 0.1, resulting in 66 different weight vectors. The control is originally discretised using 50 piecewise constant pieces of equal length. The integration and optimality tolerances employed are 10^{-6} and 10^{-8}, respectively. Again, the same initial guess has been used for all parametric SOOPs.

Figure 5 displays the 3D Pareto set obtained with the different scalarisation methods based on the convex relaxations. As can be seen, the objectives are conflicting: reducing one of the three objectives causes increases in the other two. Reducing the outlet concentration can be obtained at the expense of reducing the heat extracted from the reactor or increasing the reactor length. Alternatively, also higher net heat transfers are favoured by longer reactors. Clearly, the WS is not able to produce a nice approximation. A high number of the returned points are practically overlapping as a result of

the large difference in magnitude between the third and the first two objectives. The results for NBI and NNC exhibit a much nicer spread, although for both methods, points on the CHIM (or utopia plane) had to be selected not only as *convex* combinations of the individual minima (i.e., with all $w_i \geq 0$) but also as *non-convex* combinations (i.e., without $w_i \geq 0$) in order to avoid the overlooking of Pareto points near the boundaries. As a result, the number of SOOPs increases from 66 to 129 for the given step in the weights of 0.1. The results for NBI and NNC (incorporating the adaptation proposed in [18]) are highly similar. When both methods return a feasible solution, it is identical provided that all additional NNC inequalities are active. Also, the strict character of the additional equalities in NBI may cause the return of non-Pareto optimal points for certain weights, whereas the less stringent character of these inequalities in NNC often allows in these cases that returned points move to the boundary of the Pareto front. The Pareto filter algorithm reduces the number of final Pareto optimal solutions n_p. However, not all non-Pareto optimal points can be removed by the Pareto filter as it checks for dominated solutions only based on a finite set of candidate solutions. Nevertheless, it should be noted that in general an accurate representation of Pareto set is obtained and that the artefacts near the boundaries are in practice often of minor importance.

Figure 6 depicts the relaxed optimal jacket temperature profile and the corresponding reactor temperature profiles for the three individual minima and one intermediate solution $\mathbf{w} = [0.4, 0.3, 0.3]^{\mathrm{T}}$. The arc structure is typically *max-min-constrained-min*. In the first reactor part the reactor temperature has to be as high as possible (without violating the upper bound of 400 K) to stimulate conversion and heat production, while in the last part, it has to be decreased to recover heat. When solely conversion is aimed at, the entire reactor is exploited (i.e., $L = 1$ m) while the control is used to maintain the upper reactor temperature until the outlet (i.e., no second *min* arc). When the average net heat transfer per unit of reactor length is optimised, the optimal reactor is a little shorter ($L \approx 0.87$ m) and the upper temperature is only maintained in a tiny part, in favour of a large heat recovery section at the end (i.e., a large second *min* arc). Alternatively, the reactor length is reduced to its minimum value, when this objective is solely concentrated on. The intermediate Pareto optimal point exhibits also intermediate behaviour, as could be expected.

Clearly, the optimal relaxed solutions do not only consist of *max* and *min* arcs but also contain *path-constrained* arcs. Nevertheless, also this type of arcs (as well as *sensitivity-seeking* arcs) can be approximated to any desired accuracy by integer feasible solutions (compare [13]). Figure 7 illustrates integer feasible solutions for the minimum averaged net heat transfer case obtained after Sum Up Rounding. It has been observed that when the tolerance TOL is decreased, the number of switches increases and the 2-norm of the vector of relative deviations in the different objective functions with respect to the relaxed solution decreases from 3.4% for the top right solution, over 1.4% to 0.5% for the bottom left and bottom right solutions, respectively. However, in practice the number of switchings has of course to be limited. In summary, the Pareto set can again be approximated to any desired accuracy by integer feasible solutions, without actually solving integer problems.

4.3. Comparison with an evolutionary MO approach

The current approach is compared to a classic and successful vectorisation approach, i.e., the Non-dominated Sorting Genetic Algorithm II (`NSGA-II`) [27]. `NSGA-II` Version 1.1, that allows both real and binary variables and includes nonlinear constraint handling features, has been obtained from the Kanpur Genetic Algorithms Laboratory [28] and is coupled to the `ACADO` integrator suite [29] for simulation of the ODE systems.

In the current study no acceptable solution has been obtained for the testdrive case study. Although genetic algorithms can almost naturally incorporate integer variables, several reasons can be identified for the current inability. The MO-MIOCP has three continuous controls (i.e., the steering angle velocity, the total braking force and the accelerator pedal position) and one discrete control (i.e., the gear choice). The highly nonlinear dynamics require a fine control discretisation (number of control discretisation intervals $n_i \geq 10$). This gives already rise to 3 n_i degrees of freedom for the continuous controls and n_i for the interval lengths, which is quite high given $n_i \geq 10$. In addition, the MO-MIOCP is constrained by nonlinear inequality path constraints and terminal equality constraints.

Although the tubular reactor case involves three objectives, it can more easily be tackled by evolutionary approaches, since it only involves one integer control (i.e., the jacket fluid temperature) and is less constrained. There

is only one nonlinear inequality path constraint (Equation (27)) and one terminal inequality (Equation (30)). In the current study, the control is discretised on five intervals which can vary in length. Hence, this results in five continuous degrees of freedom and five discrete degrees of freedom that can only take five specified values. The former are implemented as real variables in `NSGA-II`, whereas the latter are each coded as binary variables described by five bits. The path constraint is incorporated via forcing the integral of the constraint violation along the reactor to be below a given small tolerance value. To enable a fair comparison, this value has been selected in the same order of magnitude as the violation encountered in the multiple shooting algorithm. The population size and the number of generations are 160 and 200, respectively. The probabilities for cross-over and mutation are 0.9 and 0.2 for the real variables and 0.9 and 0.04 for the binary variables. The resulting Pareto set is displayed in Figure 8. Clearly, the Pareto set resembles the one generated by NNC, but it is located slightly higher. As increasing the number of generations does not improve the solutions, the main reason is expected to be the lower flexibility of the control profile. Also the set of points generated by NNC exhibits visually a nicer spread along the Pareto front due to the choice of uniformly distributed points on the utopia plane.

4.4. Computational expense

To provide an indication of the computational burden, Table 1 depicts the total computation time required to generate the Pareto front. The employed computer is a regular laptop with an Intel (1.86 Ghz) processor and 2 GB RAM. n_p gives the resulting number of Pareto points after filtering.

The left column of Table 1 shows computation times in seconds for the testdrive application. The deterministic approaches were performed on a control discretisation grid with 80 intervals, making use of outer convexification. Note that the extra effort for obtaining an integer solution via Sum Up Rounding and simulation is negligible on a fixed control discretisation grid. As stated before, the stochastic approach was not able to find any feasible solution.

The weighted sum (WS) based approach results in significantly larger computation times than the NNC and NBI based methods which impose

additional constraints. The slightly smaller CPU time for NBI can be explained by the fact that equality constraints are often more easily dealt with by SQP-type of optimisation methods than inequality constraints.

The right hand side column of Table 1 shows computation times for the tubular reactor. Again the NBI approach appears to be the fastest. Despite the modest CPU for the WS, it should be noted that the accuracy of the returned Pareto set approximation is low (see Figure 5). Hence, to obtain a similar accuracy with WS, much smaller (and preferably non-uniform) steps for the weights have to be adopted, resulting a much higher number of SOOPs to be solved, and, thus, also much higher CPU times. This example has been chosen in a way that is favorable to a stochastic approach: the problem is small-scale, does not incorporate difficult point constraints, and contains controls with a discrete feasible set. Still the approach is outperformed by the proposed deterministic, gradient-based approach. Again, integer solutions can be obtained with Sum Up Rounding, examples on different control grids are shown in Figure 7.

Case study		Testdrive		Tubular reactor	
MO-MIOCP method		n_p	CPU [s]	n_p	CPU [s]
MO-MINTOC	WS	11	1436.70	66	53.18
	NNC	11	1124.90	102	90.53
	NBI	11	719.47	101	33.07
NSGA-II		—	—	160	159.59

Table 1: CPU times in seconds for testdrive (left) and tubular reactor (right) case studies. For both the NBI approach performs best. The stochastic approach `NSGA-II` is not able to produce a feasible solution for the testdrive example.

It should be stressed again that in the proposed approach an additional speed-up can be expected when hot-starts of solutions of related parametric SOOPs are used for initialisation.

4.5. Pareto sets in a first discretise, then optimise setting

In the current study a direct method based on a *first discretise, then optimise* principle has been employed. However, the approximation of the path constraint on the continuous time interval $[t_0, t_f]$ by a finite number of evaluation points (which in the current case always coincide with the multiple shooting grid) is sometimes problematic, especially if the points are allowed

to move in time. This is the case for, e.g., the testdrive example. The points in Figure 9 indicate the evaluation of the constraint. Generally speaking, whenever the accurate compliance with the path constraint is an issue, it is recommended (*i*) to include safety distances, (*ii*) to evaluate at more points, or (*iii*) to apply techniques as proposed in [30] in order to track violations. For 80 multiple shooting intervals (equivalent to evaluation points for the constraint) the violation of constraints by the trajectory is acceptable (at least to the eye) for every single subproblem. However, in the context of a Pareto set of optimal solutions nondifferentiabilities of the Pareto front may result from a change of the active set of the optimisation problem (whenever evaluation points are pushed away from the critical track passages). Although these possible nondifferentiabilities have not been explicitly taken into account, the presented approach has resulted in a satisfactory smooth Pareto set.

5. Conclusions

In the current paper a generic approach to efficiently solve MO-MIOCPs has been proposed. The rationale is a synergy between complementary approaches from the fields of continuous scalar MOO and MIOC. To tackle the multiple objective aspect, deterministic scalarisation methods as NBI and NNC have been exploited, which transform the original MO-MIOCP into a series of parametric single objective MIOCPs. These MIOCPs are then solved efficiently by deterministic direct multiple shooting techniques, which exploit convex relaxations of the integer requirements. Due to these convexification ideas, it is possible to approximate the MIOCP results at any desired accuracy in very short computation times. Moreover, if the control grid is fine enough, the Pareto set can be obtained solely based on relaxed convex formulations. The proposed approach has been illustrated successfully on a double-lane change testdrive case study which involves a detailed car model with different gears and a jacketed tubular reactor with only discrete jacket temperature levels. In the first case, the trade-off between minimising the driving time and minimising the fuel consumption has been studied, while in the second the objectives aimed at maximing the conversion and net heat recovery and minimising the reactor length. For both cases, the NBI and NNC based methods have yield an accurate approximation of the Pareto set using fine control discretisations in a very limited amount of time due to the deterministic nature and the complementarity of the procedures involved.

A comparison has been made to the evolutionary algorithm `NSGA-II`. Despite the large number of successful applications reported in literature, this algorithm was not able to generate an acceptable solution for the highly constrained testdrive case and the results for the tubular reactor case did not achieve the same accuracy as the ones obtained with the proposed approach. In summary, the current approach exhibits generic features, which should allow an easy transfer to other applications, and, hence, can pave the way for the fast and efficient solution of MO-MIOCPs in different application domains.

6. Acknowledgements

Work supported in part by Projects OT/03/30-OT/09/025/TBA, OPTEC (Center-of-Excellence Optimization in Engineering) EF/05/006 and SCORES-4CHEM KP/09/005 of the Katholieke Universiteit Leuven, and by the Belgian Program on Interuniversity Poles of Attraction, initiated by the Belgian Federal Science Policy Office, and by the Heidelberg Graduate School *Mathematical and Computational Methods for the Sciences*. J.F. Van Impe holds the chair Safety Engineering sponsored by the Belgian chemistry and life sciences federation essenscia. Permission for the usage of the software package `MUSCOD-II` by Georg Bock is gratefully acknowledged. The scientific responsibility is assumed by its authors.

Acronyms

MIOC	mixed-integer optimal control
MIOCP	mixed-integer optimal control problem
MO-MIOC	multiple objective mixed-integer optimal control
MO-MIOCP	multiple objective mixed-integer optimal control problem
MOO	multiple objective optimisation
MOOC	multiple objective optimal control
MOOCP	multiple objective optimal control problem
MOOP	multiple objective optimisation problem
NBI	normal boundary intersection
NNC	normalised normal constraint
ODE	ordinary differential equation
SOO	single objective optimisation
SOOP	single objective optimisation problem
SQP	sequential quadratic programming
WS	weighted sum

Table 2: Acronyms

References

[1] R. Sargent, Optimal control, Journal of Computational and Applied Mathematics 124 (2000) 361–371.

[2] M. Garey, D. Johnson, Computers and intractability: A guide to the theory of NP-Completeness, W.H. Freeman, New York, 1979.

[3] K. Miettinen, Nonlinear multiobjective optimization, Kluwer Academic Publishers, Boston, 1999.

[4] M. Gerdts, Solving mixed-integer optimal control problems by branch & bound: a case study from automobile test-driving with gear shift, Optimal Control Applications and Methods 26 (2005) 1–18.

[5] M. Gerdts, A variable time transformation method for mixed-integer optimal control problems, Optimal Control Applications and Methods 27 (2006) 169–182.

[6] S. Sager, M. Diehl, G. Singh, A. Küpper, S. Engell, Determining SMB superstructures by mixed-integer control, in: Proceedings OR2006, Springer, Karlsruhe, 2007, pp. 37–44.

[7] V. Bhaskar, S. Gupta, A. Ray, Applications of multi-objective optimization in chemical engineering, Reviews in Chemical Engineering 16 (2000) 1–54.

[8] I. Das, J. Dennis, Normal-Boundary Intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems, SIAM Journal on Optimization 8 (1998) 631–657.

[9] A. Messac, A. Ismail-Yahaya, C. Mattson, The normalized normal constraint method for generating the Pareto frontier, Structural and Multidisciplinary Optimization 25 (2003) 86–98.

[10] F. Logist, P. Van Erdeghem, J. Van Impe, Efficient deterministic multiple objective optimal control of (bio)chemical processes, Chemical Engineering Science 64 (2009) 2527–2538.

[11] F. Logist, J. Van Impe, Multiple objective optimisation of cyclic chemical systems with distributed parameters, in: Proceedings of the 14th

IFAC Workshop on Control Applications of Optimisation (CAO-09), 2009, CD–ROM, 6 pages.

[12] S. Sager, G. Reinelt, H. Bock, Direct methods with maximal lower bound for mixed-integer optimal control problems, Mathematical Programming 118 (1) (2009) 109–149.

[13] S. Sager, Reformulations and algorithms for the optimization of switching decisions in nonlinear optimal control, Journal of Process Control 19 (2009) 1238–1247.

[14] H. Bock, K. Plitt, A multiple shooting algorithm for direct solution of optimal control problems, in: Proceedings of the 9th IFAC world congress, Budapest, Pergamon Press, 1984.

[15] K. Deb, Multi-Objective Optimization Using Evolutionary Algorithms, John Wiley, Chichester, London, UK, 2001.

[16] I. Das, J. Dennis, A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems, Structural Optimization 14 (1997) 63–69.

[17] Y. Haimes, L. Lasdon, D. Wismer, On a bicriterion formulation of the problems of integrated system identification and system optimization, IEEE Transactions on Systems, Man, and Cybernetics SMC-1 (1971) 296?–297.

[18] J. Sanchis, M. Martinez, X. Blasco, J. Salcedo, A new perspective on multiobjective optimization by enhanced normalized normal constraint method, Structural and Multidisciplinary Optimization 36 (2008) 537–546.

[19] A. Messac, C. Mattson, Normal constraint method with guarantee of even representation of complete Pareto frontier, AIAA Journal 42 (2004) 2101–2111.

[20] S. Sager, Numerical methods for mixed–integer optimal control problems, Der andere Verlag, Tönning, Lübeck, Marburg, 2005, ISBN 3-89959-416-9. Available at http://sager1.de/sebastian/downloads/Sager2005.pdf.

[21] D. Leineweber, I. Bauer, H. Bock, J. Schlöder, An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part I: theoretical aspects, Computers and Chemical Engineering 27 (2003) 157–166.

[22] D. Leineweber, A. Schäfer, H. Bock, J. Schlöder, An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part II: Software aspects and applications, Computers and Chemical Engineering 27 (2003) 167–174.

[23] C. Kirches, S. Sager, H. Bock, J. Schlöder, Time-optimal control of automobile test drives with gear shifts, Optimal Control Applications and Methods 31 (2010) 137-153.

[24] S. Sager, MIOCP benchmark site, http://mintoc.de.

[25] F. Logist, I. Smets, J. Van Impe, Derivation of generic optimal reference temperature profiles for steady-state exothermic jacketed tubular reactors, Journal of Process Control 18 (2008) 92–104.

[26] F. Logist, P. Van Erdeghem, I. Smets, J. Van Impe, Optimal design of dispersive tubular reactors at steady-state using optimal control theory, Journal of Process Control 19 (2009) 1191–1198.

[27] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Transaction on Evolutionary Computation, 6 (2002) 181–197.

[28] Kanpur Genetic Algorithms Laboratory, http://www.iitk.ac.in/kangal/index.shtml.

[29] B. Houska, H.J. Ferreau, M. Diehl, ACADO Toolkit website. http://acadotoolkit.org.

[30] A. Potschka, H. Bock, J. Schlöder, A minima tracking variant of semi-infinite programming for the treatment of path constraints within direct solution of optimal control problems, Optimization Methods and Software 24 (2) (2009) 237–252.

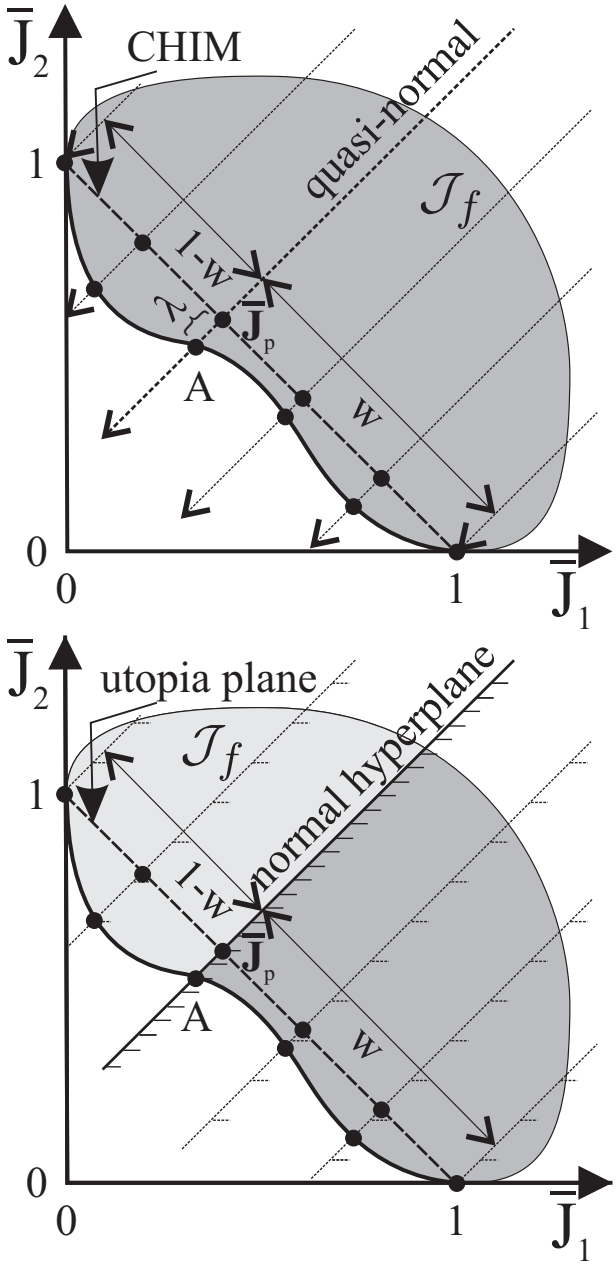

Figure 1: Schematic representation of NBI (top) and NNC (bottol) for a bi-objective problem.

Postprint version of paper published in Journal of Process Control 2010, vol. 20, pages 810-822.
The content is identical to the published paper, but without the final typesetting by the publisher.

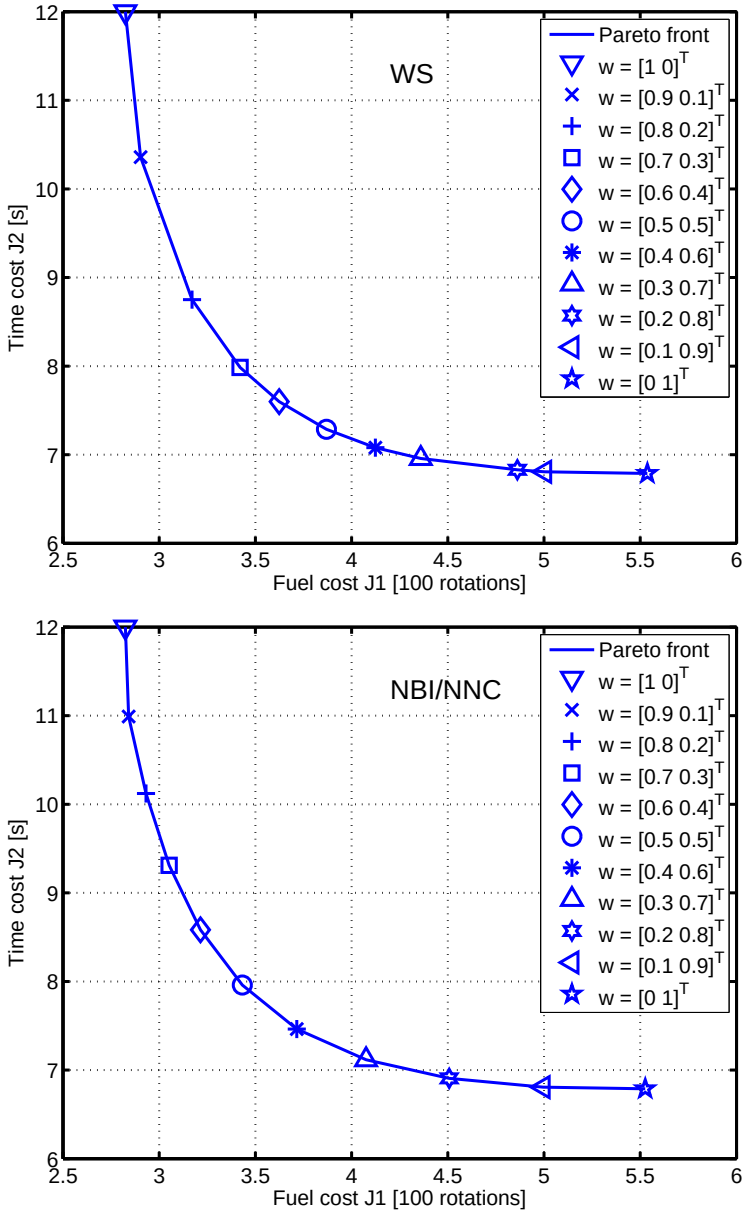

Figure 2: Testdrive: Pareto set generated by the WS (top) and NBI/NNC (bottom) approaches. (Number of shooting nodes = 80.)

Journal homepage: http://www.sciencedirect.com/science/journal/09591524
Original file available at: http://dx.doi.org/10.1016/j.jprocont.2010.04.009

Postprint version of paper published in Journal of Process Control 2010, vol. 20, pages 810-822.
The content is identical to the published paper, but without the final typesetting by the publisher.

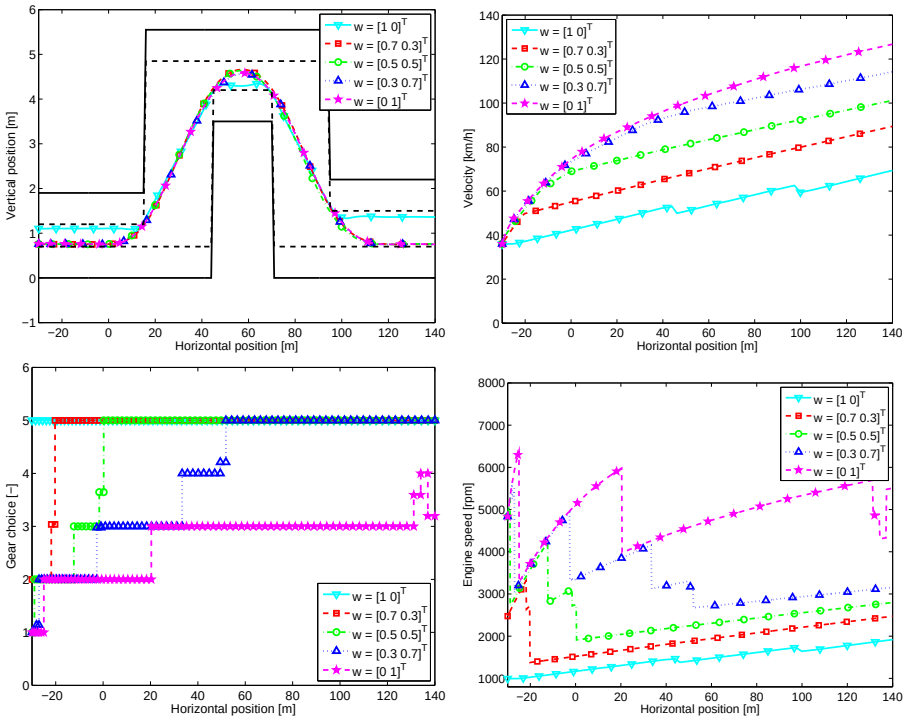

Figure 3: Testdrive: optimal trajectories (top left), velocities (top right), gear choices (bottom left), and engine speeds (bottom right) calculated with NBI and MS MINTOC. (Number of shooting nodes = 80.)

Journal homepage: http://www.sciencedirect.com/science/journal/09591524
Original file available at: http://dx.doi.org/10.1016/j.jprocont.2010.04.009

Postprint version of paper published in Journal of Process Control 2010, vol. 20, pages 810-822.
The content is identical to the published paper, but without the final typesetting by the publisher.

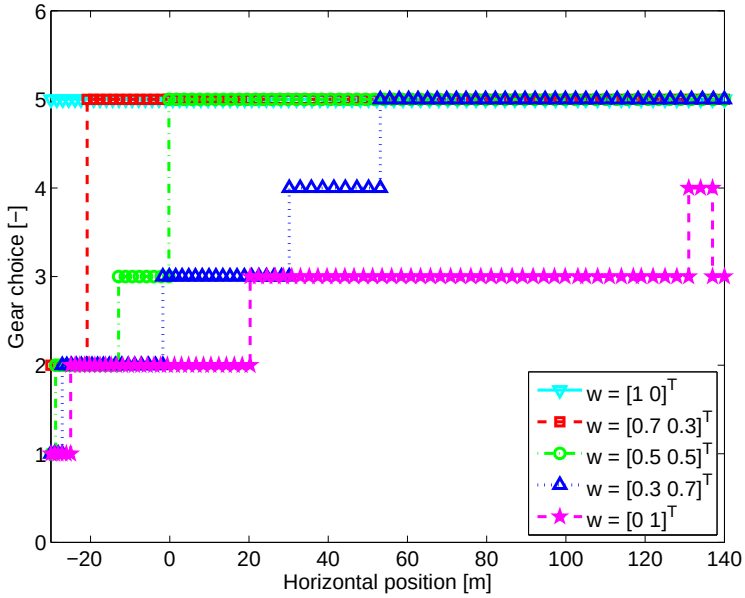

Figure 4: Testdrive: optimal integer feasible gear choices after Sum Up Rounding.

Journal homepage: http://www.sciencedirect.com/science/journal/09591524
Original file available at: http://dx.doi.org/10.1016/j.jprocont.2010.04.009

Postprint version of paper published in Journal of Process Control 2010, vol. 20, pages 810-822.
The content is identical to the published paper, but without the final typesetting by the publisher.

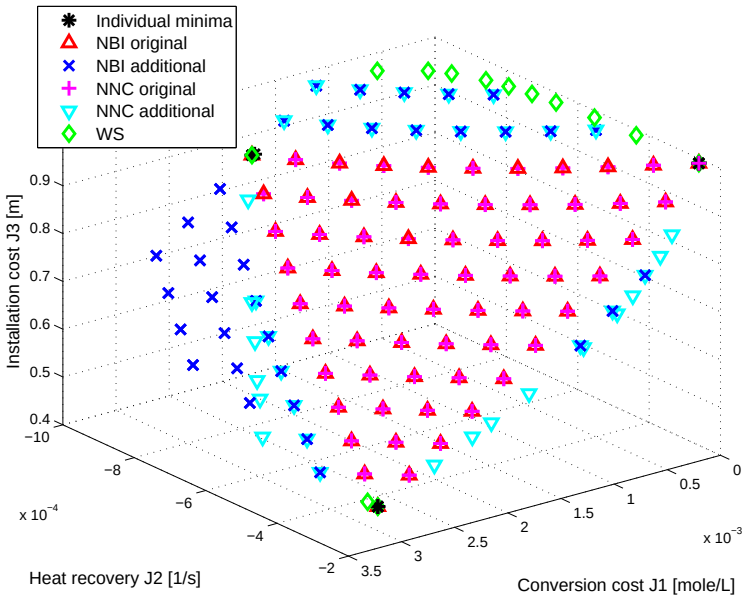

Figure 5: Tubular reactor: Pareto set generated by NBI, NNC, and WS, respectively. For NBI and NNC results are given for both the original convex weights and the additional weights generated based on the procedure reported in [19]. (Number of shooting nodes = 50.)

Journal homepage: http://www.sciencedirect.com/science/journal/09591524
Original file available at: http://dx.doi.org/10.1016/j.jprocont.2010.04.009

Postprint version of paper published in Journal of Process Control 2010, vol. 20, pages 810-822.
The content is identical to the published paper, but without the final typesetting by the publisher.

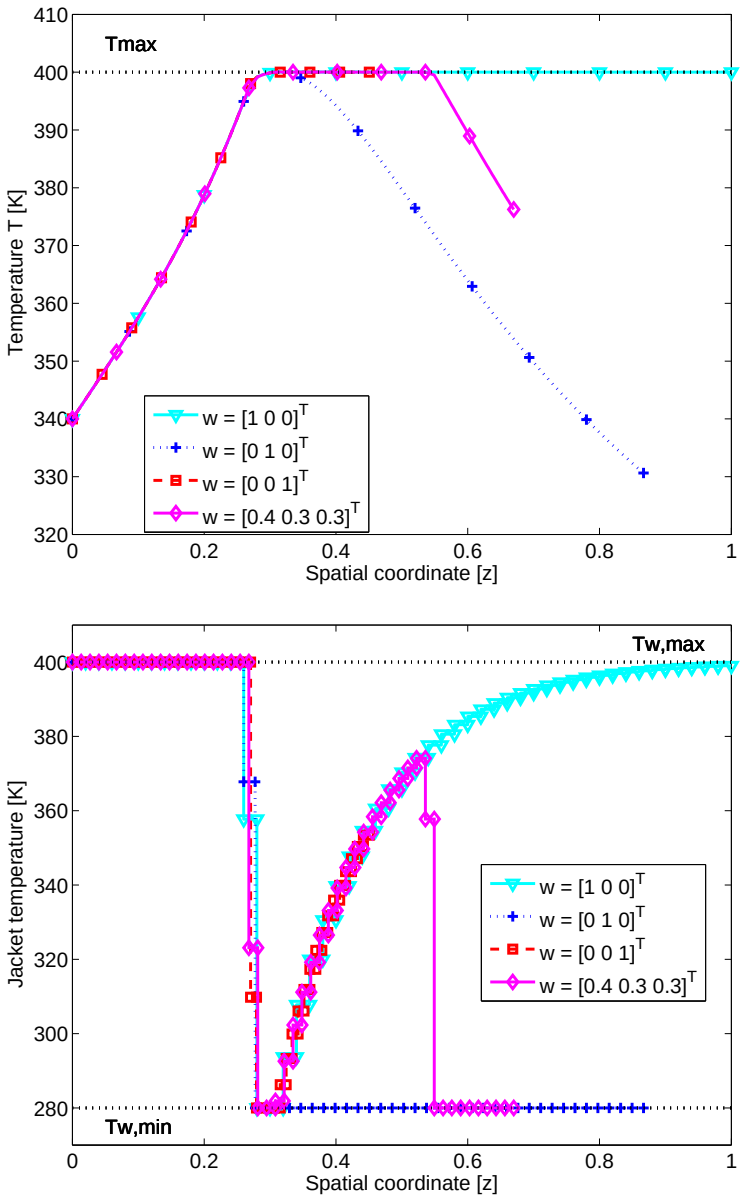

Figure 6: Tubular reactor: optimal reactor temperature (top) and (relaxed) jacket fluid temperature (bottom) profiles generated by NBI. (Number of shooting nodes = 50.) Note that the reactor length is subject to optimisation, as a result the trajectories have different lengths.

Journal homepage: http://www.sciencedirect.com/science/journal/09591524
Original file available at: http://dx.doi.org/10.1016/j.jprocont.2010.04.009

Postprint version of paper published in Journal of Process Control 2010, vol. 20, pages 810-822.
The content is identical to the published paper, but without the final typesetting by the publisher.

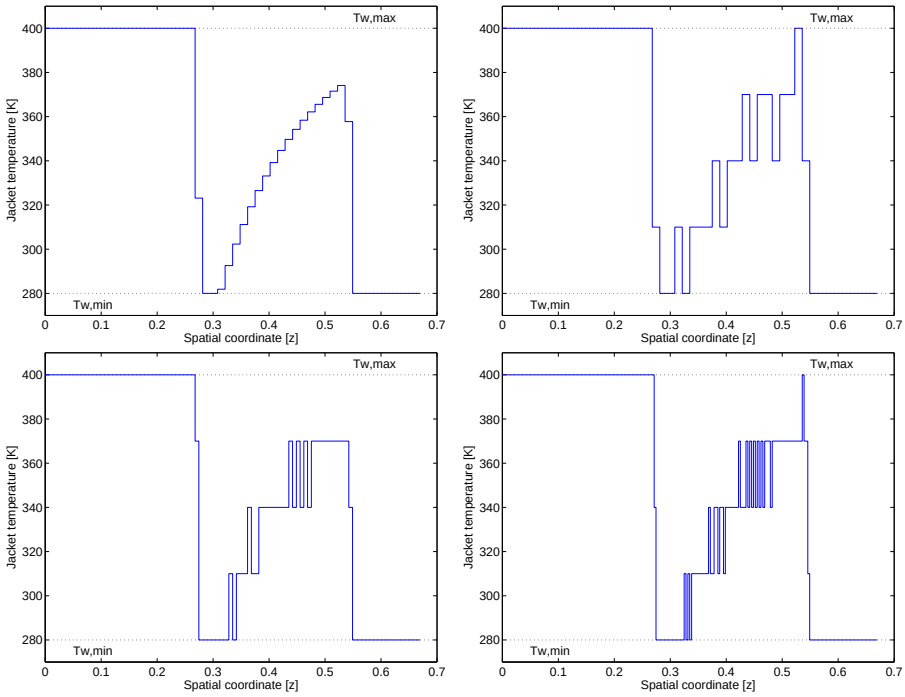

Figure 7: Tubular reactor: jacket fluid temperature control profiles for the selected setpoint with weights $\mathbf{w} = [0.4, 0.3, 0.3]^{\mathrm{T}}$. Top left: optimal relaxed solution. Others: feasible integer solutions after Sum Up Rounding on different control discretisation grids.

Journal homepage: http://www.sciencedirect.com/science/journal/09591524
Original file available at: http://dx.doi.org/10.1016/j.jprocont.2010.04.009

Postprint version of paper published in Journal of Process Control 2010, vol. 20, pages 810-822.
The content is identical to the published paper, but without the final typesetting by the publisher.

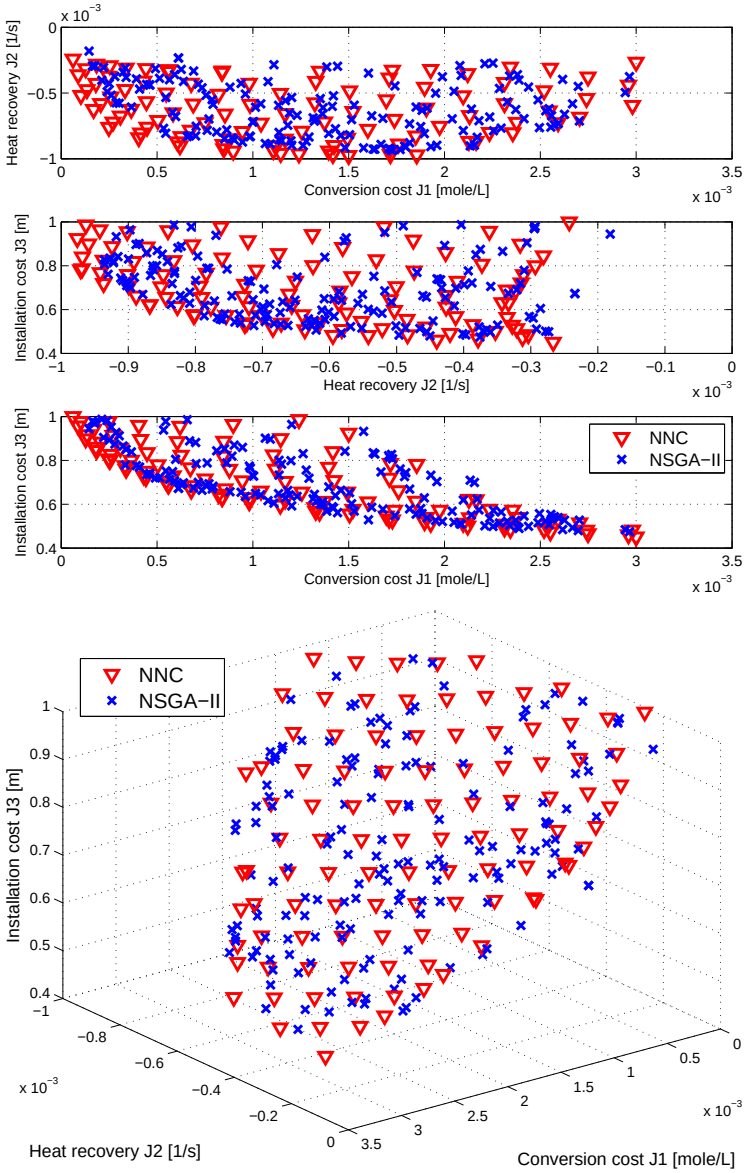

Figure 8: Tubular reactor: Pareto sets generated by NNC and `NSGA-II`: 2D (top) and 3D views (bottom).

Journal homepage: http://www.sciencedirect.com/science/journal/09591524
Original file available at: http://dx.doi.org/10.1016/j.jprocont.2010.04.009

Postprint version of paper published in Journal of Process Control 2010, vol. 20, pages 810-822.
The content is identical to the published paper, but without the final typesetting by the publisher.

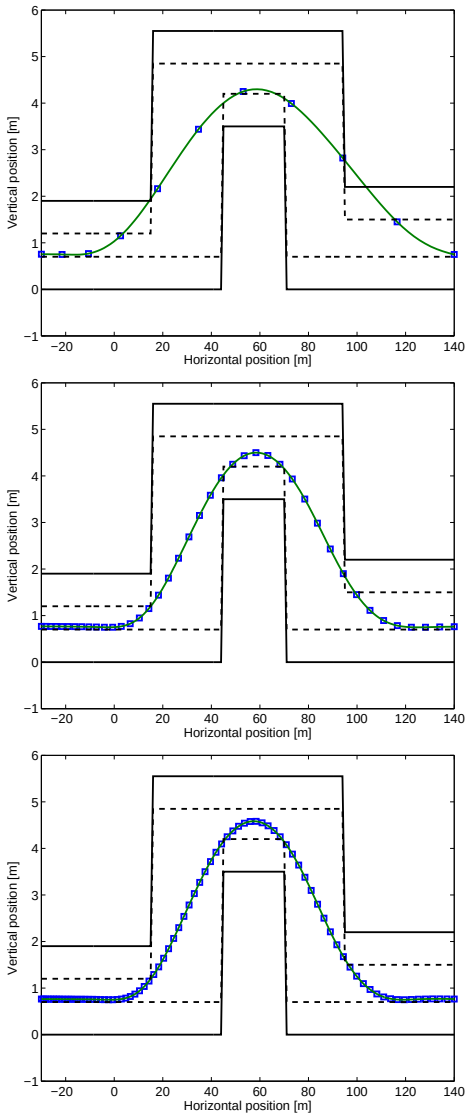

Figure 9: Testdrive: time optimal trajectories. Track (solid lines) with path constraints (dashed lines) on the vertical position for 10, 40, and 80 evaluation points, from top to bottom. Squares show evaluation points equidistant in time.

Journal homepage: http://www.sciencedirect.com/science/journal/09591524
Original file available at: http://dx.doi.org/10.1016/j.jprocont.2010.04.009